Software Based DVFS Technique for Parallel Applications to Conserve the Power

[1,] Chaitra C R, [2,] Sanjay H A, [3,] Santhosh Kumar D K

M.Tech, IV SEM, Dept. Of CSE, NMIT, Bangalore
Professor & Head, Dept. of ISE,NMIT, Bangalore
Lecturer Department of Mechatronics, Acharya Institutes, Banglore

--**ABSTRACT**---
Scientific parallel applications need more computing power, to produce such computing power we need high end hardware. This hardware consumes more power and generates lot of heat. Cooling system is required to make sure that the hardware is healthy otherwise it will be prone to failure. As a result cost of data center operation will be more and it will not be environmental friendly. By reducing the power consumption we can reduce the cost as well as make data center environmental friendly. This work proposes a software based Dynamic Voltage and Frequency Scaling technique which helps in reducing the power consumption by parallel applications. To evaluate our technique we used popular NAS parallel benchmark and observed that up to 40% energy saving is achieved.

Keywords: DVFS, Parallel application, data center, power consumption
----
Date of Submission: 07 April 2014   Date of Publication: 15 April 2014

I.   INTRODUCTION

Parallel applications are used to speed up the execution. These applications require more computing power which results in additional or complex hardware. Such hardware consumes more power. We need adequate cooling system to ensure the hardware health. It is observed that operating middle sized data center demands 80000kW power [1]. It is estimated that computing resources consume around 0.5% of the world's total power usage and it is going to increase by four times by 2020, if the current demand continues. [2]. Energy consumption by a data center is directly proportional to its operating cost. By reducing the power consumption we can reduce the operating cost.

Many solutions are proposed to reduce the power consumption by a data center. Voltage and frequency scaling are often used together to save power in mobile devices including cell phones. This technique is commonly known as DVFS, or Dynamic Voltage and Frequency Scaling. DVFS is a power management technique where the voltage and frequency used by a component is varied to conserve the power. In this technique voltage and frequency used by a system is dynamically varied using a hardware set up. In case of parallel applications which require complex hardware such as cluster, which consists of two are more interconnected systems, varying the frequency and voltage through hardware is not a feasible solution. If the requirements keep changing it becomes very difficult to tune the hardware components manually to make the system run in desired voltage and frequency and also maintenance is very difficult.

We propose software based DVFS technique for parallel applications to overcome the problem of hardware based DVFS technique. It also reduces additional hardware cost which is required to vary the frequency and voltage scaling. In our technique we will dynamically scale the voltage and frequency using the software when a parallel application is running on several nodes of the cluster to reduce the power consumption. We have used the class C NAS parallel benchmark applications [3] to test our technique.

1.2Voltage Scaling
Voltage scaling is achieved using voltage layer and regulatory framework (driver). When the CPUFreq driver scales the device frequency, voltage corresponding to the frequency (target_voltage) is looked-up in the opp list. The device scale function requests the voltage layer to scale the device voltage to the target voltage

1.2.1 Voltage Layer

The voltage layer consists of the information of all voltage domains in the system and configures all vdds during voltage layer initialization. When a vdd is configured a regulator supply handle is acquired and stored in the corresponding vdd structure. The regulators scale/set voltage function is plugged in to the vdd's voltage scale function pointer. Thus when a voltage change is requested it will be forwarded to a vdd. The voltage layer requests the regulatory framework to change the device voltage to the given voltage. Regulator driver verifies if the given voltage is within the limits of the voltage domain and regulator supply constraints. If all the checks are successful then regulator changes the voltage of the requested device to the given voltage.

Rest of the paper is organized as follows section 2 briefs about the related work; section 3 explains software based DVFS; section 4 talks about the experimental setup and results, followed by conclusion.

II.   RELATED WORKS

There are couple of efforts on DVFS technique. [3] Concentrates on applying DVFS technique on communication links in an embedded system. It proposes a scheduling algorithm for real time applications that captures both data flow and control flow of information.

In [4] a program execution time is decomposed into two parts: on-chip and off-chip. The CPU voltage/frequency is scaled based on the ratio of the on-chip and off-chip latencies for each process. [5] Proposes an algorithm which regularly updates the voltage and frequency of the processor. [6] Investigates the problem of considering DVS processor power variations dependent on tasks, during the synthesis of distributed embedded systems and its impact on energy savings. [8] Presents a power-efficient variable-voltage scheduling algorithm.

All the previous discussed works concentrate on hardware based DVFS technique which requires additional hardware to dynamically vary the voltage and frequency. Most of the works consider the embedded systems. In our work we propose a software based DVFS technique for parallel applications.

III.   PROPOSED SYSTEM

Scientific parallel applications consume more power since they need higher computing power. To reduce the power consumption DVFS Technique can be used which allows the user to vary frequency and voltage of the system as per the requirement. In this work we vary the frequency and voltage of the processor using software tools.

Benchmark Application	Number of Nodes	Frequency	Time Consumed	Power in Watts	Energy in Joules
MG	2	2 GHz	14416.10	157	3
		2.67GHz	50223.65	210	4.28
		DVFS	64234.17	173	3.4
CG	2	2 GHz	318.49	157	0.8
		2.67GHz	294.57	210	1
		DVFS	297.38	210	1
EP	2	2GHz	297.38	157	1
		2.67GHz	266.45	210	0.7
		DVFS	206.79	210	0.7
LU	2	2GHz	1701.33	157	4.4
		2.67GHz	1699.63	210	6
		DVFS	1699.60	210	6

To vary the frequency of the system we have written the program which makes use of CPUFreq module. The application running on the system is monitored. Based on the usage criteria frequency is change. For changibng it we need to get into userspace that can be done by changing the scaling governor. Above can be achieved by the command "echo user space > /sys/devices/system/cpu/cpu*/scaling_governor". Once we are in userspace we simply supply the frequency in MB to the system using command " echo <frequency> > /sys/devices/system/cpu/cpu*/scaling_setspeed"

3.1 DVFS Model:

The dynamic power consumption of a power processor is proportional to the product of Thermal design power (i.e TDP), frequency (f) , the square of the supply voltage(v^2)

$P= T\ V^2 f$................. (1)

Energy consumption is reduced by lowering the average power (P_{avg}) consumed for some duration or delay (D=$t_2 - t_1$)

$E = P_{avg} * (t_2 - t_1) = P_{avg} * Delay$...........(2)

The formulas we are using are taken from [9]

3.2 Frequency Scaling

Frequency scaling is achieved using CPUFreq framework. CPUfreq is a Linux kernel framework. It monitors the performance requirements of a processor(s) and takes decision to increase or decrease operating frequency in order to save power and/or reduce leakage power. CPUFreq consists of two elements

- ✓ The Governor - that makes decisions
- ✓ The Driver - acts based on the decisions made by the governor

IV.   EXPERIMENTAL SETUP AND RESULTS

Our experimental setup consists of 2 machines, each having a Intel(R) Core(TM) 2 Duo CPU E8200 @ 2.66GHz. A cluster was setup between these 2 machines using MPI library. Before we carry on this setup we need to check if the processor is DVFS enabled which can be know by checking in BIOS settings. If it is not enabled, enable it by turing on SpeedStep in Intel Processor or AMD Cool'n'Quiet or☐ VIA LongHaul options in BIOS settings of respective processors. We performed experiments with several different NAS parallel benchmark applications, the results of which are as follows. NAS benchmarks are used to for performance evaluation of the systems.

The benchmarks used all belongs to C class. Class refers to the size of the problem.From the above results, we can see that MG benchmark is both computation and communication intensive application. Our design technique scales frequency depending on the CPU utilization. This benchmark sometimes uses very less CPU and sometimes consumes 100% of CPU. So depending on the utilization frequency is scaled thus optimizing the energy.

The other benchmarks like CG, EP and LU run only for few minutes even though they belong to class C. And also all of them are CPU intensive applications. Most of the time the application uses 100% CPU. From the experimental results we can observe that even though the applications run at lower frequency the CPU will complete the task in about same time as when it runs at higher frequency. Therefore we can run CPU intensive applications at lower frequency itself. Our design strategy scales up the frequency once CPU utilization goes beyond 80%. Therefore the application runs at higher frequency so not much energy optimization is seen for those applications

The graph in fig.1 shows the power consumption of MG benchmark using DVFS technique when compared with other static frequencies the processor provides.

Fig 1 shows the Power variations of MG benchmark when run at different frequencies .From the above figure, we can see that as the frequency of the system increases power consumption by the application also increases but time decreases. When we run the system using DVFS technique we can see that time consumed is also reduces and also power which further reduces energy consumption which is shown in fig 2
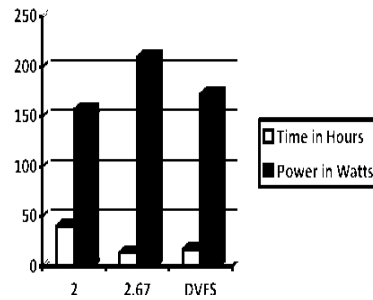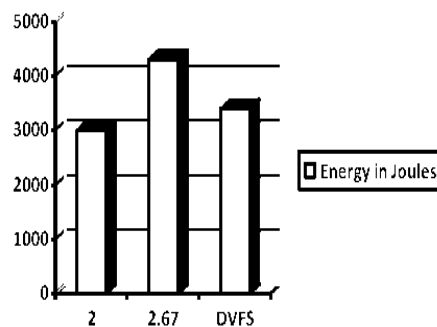
Fig 1: Power variation of MG benchmark application

Fig 2: Energy consumption reduction achieved when DVFS technique is used

V.   CONCLUSION

By reducing the power consumption we can reduce the operating cost of a data center. There are several techniques to reduce the power consumption. DVFS is one of such technique. We propose a software based DVFS technique to reduce the power consumption by parallel applications. In our work we dynamically scale the voltage and frequency from the software to reduce the power consumption. We observed that 40% energy saving is achieved in case of class C NAS parallel bench mark applications using our technique.

DVFS technique is an efficient tool that is provided by operating system which enables the user to vary the frequency and voltage of the system as per his/her requirement. We have worked on varying the processor frequency using a software approach. With this, certainly energy saving is achieved but we can achieve still more energy saving by varying the voltage of the processor. There are ways of varying the voltage using hardware tools but   our aim was to vary the voltage using software. We further try to save energy consumption varying the voltage dynamically using a software tool which helps in further reduction of energy

REFERENCES

[1]. L. Wang, G. von Laszewski, J. Dayal, X. He, and T. R. Furlani, "Thermal Aware Workload Scheduling with Backfilling for Green Data Centers," in Proceedings of the 28th IEEE International Performance Computing and Communications Conference , Arizona, U.S., Dec 2009.

[2]. W. Forrest, "How to cut data centre carbon emissions?" Website, December 2008. [Online].Available:http://www.computerweekly.com/Articles/2008/12/05/233748/how-to-cut-data-centre-carbon-emissions.html

[3]. Mostafa E.Salehi, Mehrzad Samadi, Mehrdad Najibi, Ali Afzali-Kusha, Masoud Pedram and Sied Mehdi Fakhraie,"Dynamic Voltage and Frequency Scheduling for Embedded Processors Considering Power/Performance Tradeoffs",IEEE,2010

[4]. Liong Luo, Niraj K.Jha,"Simultaneous Dynamic Voltage scaling of processors and communication Links in Real time Distributed Embedded Systems",IEEE,2007

[5]. Kihwan Choi, Ramakrishna Soma, Massoud Pedram, "Dynamic voltage and frequency Scaling based on workload Decomposition",ISLPED,2004

[6]. Mostafa E.Salehi, Mehrzad Samadi, Mehrdad Najibi, Ali Afzali-Kusha, Masoud Pedram and Sied Mehdi Fakhraie,"Dynamic Voltage and Frequency Scheduling for Embedded Processors Considering Power/Performance Tradeoffs",IEEE,2010

[7]. Yumin Zhang, Xiaobo Hu, Danny Z. Chen,"Task scheduling and voltage selection for energy minimization", DAC,2002

[8]. Marcus T.Schmitz, Bashir M. Al-Hashmi, "Considering power variations of DVS PEs for energy minimisation in Distributed systems ",ISSS,2001

[9]. Rong Ge, Xizhou, Kirk /Cameron, "Performance Constrained Distributed DVS Scheduling for Scientific Applications on Power-aware Clusters"

[10]. http://www.linux-phc.org

[11]. http://www.nas.nasa.gov/publications/npb.html